Cika Electrónica S.R.L.
Comentario Técnico: CTC-072
Título: **Utilización de XBee Wi-Fi para sensores remotos**
Autor: Sergio R. Caprile, Senior Engineer

Revisiones	Fecha	Comentarios
0	23/09/11	

En este comentario técnico estudiaremos la forma de configurar los módulos XBee Wi-Fi para utilización con aplicaciones de sensores remotos.

Introducción

En los casos que veremos a continuación, el módulo transmitirá la información de estado de sus entradas y salidas a uno o varios hosts o módulos XBee Wi-Fi remotos. Las formas de configuración para envío de información en las diversas topologías han sido descriptas en CTC-071.
Cada uno de los pines DIO*x* que puede ser configurado para su operación como entrada analógica, entrada digital, o salida digital, posee un comando *ATDx* que permite operar sobre esta selección. Los pines DIO10 a DIO12 se configuran mediante los comandos *ATP0* a *ATP2*, respectivamente.

Entradas analógicas

Para configurar alguna de las entradas como analógica, es decir, asociada al conversor analógico a digital, utilizamos el comando que le corresponde, por ejemplo *ATD2=2* setea el pin AD2/DIO2 como AD2, es decir, asociado al ADC. Los módulos poseen una referencia interna de 1,25 ó 2,5V cuyo valor puede configurarse mediante el comando *ATAV*. No debe conectarse el pin VREF dado que no se lo utiliza.

Entradas digitales

Para configurar alguna de las entradas como digital, utilizamos el comando que le corresponde, por ejemplo *ATD0=3* setea el pin AD0/DIO0 como entrada (DI). El comando *ATP1=3* setea el pin DIO11 como entrada (DI).

Lectura local de la información

Es posible leer el estado de los pines desde la UART, mediante el comando *ATIS*. Este comando fuerza una muestra inmediata, y devuelve por la UART la información. En el modo AT, dicha información estará en ASCII:

```
+++OK
atis
01
081C
03
0018
0250
021E
```

Este método en realidad sólo es confiable si no tomamos muestras automáticamente, dado que si es así, en realidad estamos tomando muestras de ese proceso de muestreo. El primer número que observamos es la cantidad de muestras que hay en el buffer, el segundo es la configuración de canales digitales y el tercero la de canales analógicos; el siguiente es el estado de las entradas y salidas digitales habilitadas, y sólo está presente si las hay. A continuación, los valores de todos los canales AD habilitados.

Si utilizamos el modo API, deberemos enviar el comando dentro de una trama de comando y recibiremos la respuesta dentro de una trama de respuesta. El formato de las tramas se describe en el manual del usuario, un ejemplo de como generar tramas API ha sido descripto en CAN-095. Por ejemplo, si no estamos trabajando en el modo con caracteres de escape, el siguiente mensaje:

```
7E 00 04 08 12 49 53 49
```

solicita una muestra.

`7E`: inicio de trama
`00 04`: longitud de datos (4 bytes a continuación, más uno de checksum)
`08`: comando
`12`: identificador de trama (cualquier número distinto de cero)
`49 53`: IS, el comando
`49`: checksum

Ejecutada la acción, recibiremos una trama conteniendo el resultado:

```
7E 00 0F 88 12 49 53 00 01 00 1C 03 00 18 02 28 02 1E 47
```

`7E`: inicio de trama
`00 0F`: longitud de datos (15 bytes a continuación, más uno de checksum)
`88`: respuesta a comando
`12`: identificador de trama (el valor que enviamos en el comando)
`49 53`: IS, el comando enviado
`00`: resultado, OK
`01`: cantidad de muestras
`00 1C`: I/O habilitados (DIO4, DIO3, DIO2)
`03`: canales analógicos habilitados (AN1, AN0)
`00 18`: estado de I/O
`02 28`: valor de AN0
`02 1E`: valor de AN1
`47`: checksum

Lectura remota de la información

Es posible solicitar a un módulo que pida la ejecución del comando *ATIS* a un módulo remoto. Para esto, el módulo XBee Wi-Fi que controla debe estar en modo API, dado que la solicitud de comandos remotos se realiza mediante una trama especial en la que se indica la dirección del módulo sobre el que se opera, y el comando en sí.

Por ejemplo, si no estamos trabajando en el modo con caracteres de escape, el siguiente mensaje:

```
7E 00 0D 07 12 00 00 00 00 C0 A8 01 10 00 49 53 D1
```

solicita al módulo con dirección 00 13 A2 00 40 0A 5E 3B que tome una muestra.

`7E`: inicio de trama
`00 0D`: longitud de datos (13 bytes a continuación, más uno de checksum)
`07`: comando remoto
`12`: identificador de trama (cualquier número distinto de cero)
`00 00 00 00 C0 A8 01 10`: dirección IP del destinatario (192.168.1.16)
`00`: opciones
`49 53`: IS

`D1`: checksum

Ejecutada la acción, recibiremos una trama conteniendo el resultado:

```
7E 00 17 87 12 00 00 00 00 C0 A8 01 10 49 53 00 01 08 1C 03 00 18 02 5B 02
1E 94
```

`7E`: inicio de trama
`00 17`: longitud de datos (23 bytes a continuación, más uno de checksum)
`87`: respuesta a comando remoto
`12`: identificador de trama (el valor que enviamos en el comando)
`00 00 00 00 C0 A8 01 10`: dirección IP del destinatario (192.168.1.16)
`49 53`: IS, el comando enviado
`00`: resultado, OK
`01`: cantidad de muestras
`08 1C`: I/O habilitados (DIO11, DIO4, DIO3, DIO2)
`03`: canales analógicos habilitados (AN1, AN0)
`00 18`: estado de I/O
`02 5B`: valor de AN0
`02 1E`: valor de AN1
`94`: checksum

También es posible solicitarlo desde un host utilizando el XBee Application Service, un ejemplo de como utilizarlo ha sido descripto en CAN-096.

Por ejemplo, enviando el siguiente mensaje al port 0xBEE de un módulo:

```
02 00 5A 00 49 53
```

solicita a dicho módulo que tome una muestra.

`02`: comando
`00`: opciones
`5A`: identificador de trama (cualquier número distinto de cero)
`00`: opciones
`49 53`: IS

Ejecutada la acción, recibiremos un mensaje conteniendo el resultado:

```
82 00 5A 49 53 00 01 00 1C 03 00 18 03 FF 03 FF
```

`82`: respuesta a comando
`00`: opciones
`5A`: identificador de trama (el valor que enviamos en el comando)
`49 53`: IS, el comando enviado
`00`: resultado, OK
`01`: cantidad de muestras
`00 1C`: I/O habilitados (DIO4, DIO3, DIO2)
`03`: canales analógicos habilitados (AN1, AN0)
`00 18`: estado de I/O
`03 FF`: valor de AN0
`03 FF`: valor de AN1

Envío periódico de la información

En el caso más común, nos interesa enviar la información de forma periódica a algún módulo o host remoto. El comando *ATIR* nos permite setear la frecuencia de muestreo en milisegundos, por ejemplo *ATIR=3E8* configura al módulo para tomar muestras cada 1s (recordemos que los valores son en hexadecimal).

Recepción de la información

Si recibiremos la información de I/O en un módulo, deberemos tener a éste en modo API. La información proveniente del otro módulo se envía por la UART en una trama de I/O. A menos que explícitamente estemos trabajando en el modo con caracteres de escape, tendremos algo como lo siguiente:

```
7E 00 15 82 00 00 00 00 C0 A8 01 10 34 00 01 00 1C 03 00 18 02 15 02 1D 62
```

`7E`: inicio de trama
`00 15`: longitud de datos (21 bytes a continuación, más uno de checksum)
`82`: recepción de datos de I/O
`00 00 00 00 C0 A8 01 10`: dirección IP de quien envía (192.168.1.16)
`34`: RSSI
`00`: opciones
`01`: cantidad de muestras
`00 1C`: I/O habilitados (DIO4, DIO3, DIO2)
`03`: canales analógicos habilitados (AN1, AN0)
`00 18`: estado de I/O
`02 15`: valor de AN0
`02 1D`: valor de AN1
`62`: checksum

Si recibimos la información de I/O en un host IP, recibiremos en el port 0xBEE algo como lo siguiente:

```
04 00 01 00 1C 03 00 18 02 22 02 1E
```

`04`: recepción de datos de I/O
`00`: no aplica
`01`: cantidad de muestras
`00 1C`: I/O habilitados (DIO4, DIO3, DIO2)
`03`: canales analógicos habilitados (AN1, AN0)
`00 18`: estado de I/O
`02 22`: valor de AN0
`02 1E`: valor de AN1

Esto corresponde al XBee Application Service, un ejemplo de como utilizarlo ha sido descripto en CAN-096.

Bajo consumo

Si el módulo se configura para bajo consumo, al salir de su letargo, si tiene configurado algún valor de *ATIR*, toma una muestra y la envía. Si la señal a medir no está lista al momento de despertar, la medición puede ser errónea, por ejemplo si apagamos una referencia de tensión externa para minimizar aún más el consumo. En este caso, es conveniente configurar una frecuencia de muestreo que permita tomar más de una muestra y descartar la primera. El módulo sigue tomando muestras mientras está despierto.

La forma de operar en bajo consumo puede configurarse de modo similar a como hiciéramos en CTC-071, aunque en el caso de un sensor remoto sin ninguna otra aplicación, podemos aprovechar la facilidad de *cyclic sleep* del módulo. El comando *ATSM=4* habilita el modo *cyclic sleep*. En estas condiciones, la corriente consumida es: $I=\frac{I_{Rx}\cdot t_{on}+I_{sleep}\cdot t_{off}}{t_{on}+t_{off}}$. Si bien la diferencia de consumo en transmisión es apreciable, el

tiempo que dura es muy pequeño en comparación y nos permite simplificar el cálculo. Los valores de t_{on} y t_{off} dependen del modo de operación.

AP Associated Sleep

El tiempo que el módulo permanece en actividad ronda los 30ms. Si durante ese tiempo detecta actividad en el puerto serie, permanece despierto por un tiempo que se configura mediante el comando *ATST*. Seleccionamos este modo configurando *ATSO=40*.
Si se configura *ATIR* > 0, el módulo toma y transmite muestras mientras está despierto.
El tiempo que el módulo permanece encendido es: $t_{on}=t_{ST}\geqslant 30\text{ms}$, y el que permanece apagado es $t_{off}=t_{beacons}-t_{on}$, asumiendo que si el valor de ST es alto, sólo dormirá entre beacons cuando no haya actividad en el puerto serie. Con el valor típico de 100ms entre beacons, el consumo del módulo es cercano al 30%, es decir, unos 50mA.

Deep Sleep

El tiempo que el módulo permanece en actividad se configura mediante el comando *ATST*. El tiempo que no está activo se indica mediante el comando *ATSP*. Por ejemplo, seteando *ATSO=0*, *ATSP=1000*, *ATIR=1000* y *ATST=2000*, el módulo permanece en bajo consumo por 40,96 segundos, despierta, se asocia, toma una muestra y la transmite, espera por actividad en la interfaz, y se va a dormir, repitiendo el ciclo. El tiempo que está despierto no supera los 8,192s, aún si el módulo no logra asociarse o hay actividad en la interfaz serie.
Si se configura *ATIR* > *ATST*, el módulo toma y transmite sólo una muestra.
El tiempo que el módulo permanece encendido es: $t_{on}=t_{ST}>t_{assoc}+t_{Tx}$, y el que permanece apagado es $t_{off}=t_{SP}$.